

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET
CÂMPUS GUARAPUAVA

LUCAS ANDREY GROXKO JACOMEL

SISTEMA WEB DE BIBLIOTECAS ESCOLARES

PROJETO DE TRABALHO DE CONCLUSÃO DE CURSO DO CURSO SUPERIOR
EM TECNOLOGIA EM SISTEMAS PARA INTERNET

GUARAPUAVA
1º Semestre de 2017

LUCAS ANDREY GROXKO JACOMEL

Sistema Web de bibliotecas escolares

Projeto de Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 1, do Curso Superior de Tecnologia em Sistemas para Internet – TSI – da Universidade Tecnológica Federal do Paraná – UTFPR – Câmpus Guarapuava, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador (a): Prof. Dr. William Alberto Cruz Castañeda.

GUARAPUAVA
1º Semestre de 2017

RESUMO

Jacomet, Lucas Andrey Groxko. BiblioEscola : Sistema web de bibliotecas escolares. 37f. Projeto do Trabalho de Conclusão de Curso – Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava 2017.

O município de Guarapuava possui 57 escolas da rede pública, localizadas dentro do perímetro urbano, sendo que 22 delas são estaduais e as 35 restantes municipais. Em relação a infraestrutura bibliotecária destas escolas, existem locais que não possuem sistema informatizado devido aos altos custos de aquisição e manutenção. Desta forma, o conceito de biblioteca eletrônica permite colocar à disposição dos usuários um sistema no qual os processos básicos são de natureza eletrônica e automatizados. O objetivo desse trabalho é desenvolver um sistema *web* de biblioteca escolar que atenda os requisitos do sistema bibliotecário de uma escola pública do município de Guarapuava. Para o desenvolvimento deste trabalho utiliza-se a metodologia *Rational Unified Process* (RUP) dividido em quatro etapas sendo elas a concepção, elaboração, construção e testes. Este trabalho contribuirá para que as escolas da rede pública de Guarapuava modernizem e organizem melhor seus processos bibliotecários.

Palavras-chave : sistema web, bibliotecas, escola pública.

ABSTRACT

Jacomel, Lucas Andrey Groxko. BiblioEscola: web system to school library. 37p. Project of the Course Completion Work - Technology Course in Internet Systems, Federal Technological University of Paraná. Guarapuava 2017.

The municipality of Guarapuava has 57 public schools within the urban perimeter, in which 22 are state and 35 are municipal. Regarding the librarian infrastructure of these schools, there are places that do not have computerized system due to high acquisition and maintenance costs. Thus, the concept of electronic library allows to make available to users with a system in which basic processes are electronic and automated nature. The objective of this work is to develop a web system of school library that meets the requirements of a librarian system of a public school in the municipality of Guarapuava. For the development of this work, the Rational Unified Process (RUP) methodology is used and divided into four stages: design, elaboration, construction and testing. This work will help to modernize and organize Guarapuava's public schools in their librarian processes.

Key words: web system, library, public school.

LISTA DE FIGURAS

Figura 1 – Módulo de catalogação do OpenBiblio.....	14
Figura 2 – Tela de cadastro de leitores no BibLivre.....	15
Figura 3 – Tela de pendências do Meu Pergamum por usuários.....	16
Figura 4 – Como funciona a arquitetura cliente-servidor.....	18
Figura 5 - Diagrama de casos de uso que modelam o cenário 1.....	30
Figura 6 - Diagrama de casos de uso que modelam o cenário 1.....	31
Figura 7 - Diagrama de casos de uso que modelam o cenário 1.....	31
Figura 8 - Diagrama de casos de uso que modelam o cenário 2.....	32
Figura 9 - Diagrama de casos de uso que modelam o cenário 3.....	32
Figura 10 - Diagrama de casos de uso que modelam o cenário 4	33
Figura 11 - Diagrama de casos de uso que modelam o cenário 5.....	33
Figura 12 - Diagrama de casos de uso que modelam o cenário 6.....	34

LISTA DE QUADROS

Quadro 1 Cronograma de desenvolvimento do projeto.....36

LISTA DE SIGLAS

TIC	Tecnologia da Informação e Comunicação
RUP	Rational Unified Process
SAB	Sistema de automação bibliotecário
PHP	Hypertext PreProcessor
SQL	Structured Languagem Query
PUC-PR	Pontifícia Universidade Católica do Paraná
IP	Internet Protocol
JEE	Java Enterprise Edition
SE	Standart Edition
API	Application Programming Interface
JSP	Java Server Pages
JSF	Java Server Faces
EJB	Enterprise Java Beans
JHTML	Java HyperText Markup Language
GUI	Grafical User Interface
JPA	Java Persistence API
UML	Unified Modeling Language
CRUD	Create Read Update Delete

SUMÁRIO

1. Introdução	10
1.1 Objetivos	11
1.1.1 Objetivo Geral.....	11
1.1.2 Objetivos Específicos.....	11
2 Estado da arte.....	12
2.1 Openbiblio.....	14
2.2 Biblivre.....	15
2.3 Pergamum.....	15
2.3.1 Implantação do pergamum na rede de biblioteca da prefeitura de Vitória-ES.....	17
3 Fundamentação Teórica.....	17
3.1 Arquitetura Cliente-Servidor.....	17
3.2 Java Enterprise Edition – JEE.....	18
3.2.1 Java Server Pages – JSP.....	19
3.2.2 Java Servlets.....	19
3.2.3 Java Server Faces – JSF.....	19
3.2.4 Enterprise Java Beans – EJB.....	20
3.2.5 Java Persistence API – JPA.....	20
3.3 Apache Tomcat Server.....	20
3.4 Servidor de banco de dados MySQL.....	20
3.5 Unified Modeling Language (UML).....	21
3.5.1 Diagramas de casos de uso.....	21
3.5.2 Diagrama de classes.....	22
3.6 Framework – Hibernate.....	22
4 Procedimentos Metodológicos/Metodologia.....	23
5 Desenvolvimento Preliminar.....	24
5.1 Especificação de requisitos.....	24
5.2 Requisitos funcionais do sistema.....	27
5.3 Requisitos não-funcionais do sistema.....	29
5.4 Principais características e funções do sistema.....	29

6 Resultados esperados.....	30
7 Considerações Finais.....	35
8 Cronograma.....	36
9 Referências.....	37

1. Introdução

No mundo globalizado e na sociedade da informação, cujas tendências se sustentam pela acelerada inclusão de tecnologias da informação e comunicação (TICs), podemos perceber uma nova realidade em todos os setores e organizações da sociedade.

Especialmente no setor educacional público, as mudanças por essa inclusão das TICs criaram alterações nos hábitos de uso da informação de setores como o bibliotecário. Impulsionando a organização para a busca de um processo de modernização de suas estruturas e maior agilidade na prestação de serviços à comunidade escolar.

No contexto local do setor educacional público de Guarapuava, a cidade possui 57 escolas da rede pública, sendo 22 delas estaduais e as 35 restantes municipais. Porém existem algumas escolas desta rede que não possuem sua sede própria e dividem a infraestrutura com uma outra. Além disso, cada uma tem a sua biblioteca separada e não possuem um sistema de gerenciamento dos serviços biblioteconômicos.

O desenvolvimento deste trabalho está sendo dividido e executado em quatro etapas baseadas na metodologia *Rational Unified Process* (RUP), contribuindo para que as escolas da rede pública de Guarapuava modernizem e organizem a sua biblioteca.

1.1 Objetivos

1.1.1 Objetivo Geral

Desenvolver um sistema de gerenciamento *web* de bibliotecas para escolas públicas na cidade de Guarapuava.

1.1.2 Objetivos Específicos

- Obter e estabelecer requisitos para o sistema;
- Estruturar os módulos do sistema;
- Estruturar o banco de dados do Sistema;
- Implementar o banco de dados;
- Desenvolver os módulos do sistema;
- Integrar módulos do sistema;
- Validar o projeto de TCC;

2. Estado da arte

A necessidade de estar atualizado e envolvido com avanços tecnológicos é de extrema importância para profissionais das escolas da rede pública, como, por exemplo, os bibliotecários. Se antes ficavam restritas aos limites físicos de uma biblioteca e de uma coleção, agora o uso difundido da tecnologia a serviço da informação transpõe barreiras físicas e institucionais (da Silva, et. al, 2010).

Além da biblioteconomia, áreas ligadas à informação como a arquivologia e a ciência da computação trabalham em conjunto, já que estão intimamente relacionadas à construção, fluxo, organização e armazenagem das informações. (da Silva, et. al, 2010).

Os sistemas desenvolvidos baseados neste perfil passam a ser chamados de Sistemas de Automação Bibliotecários (SABs). Os SABs são sistemas de bases de dados com uma finalidade específica, projetados para controlar atividades de uma biblioteca (Lima, 1999).

Os SABs baseados na filosofia *open-source* apresentam como um dos principais motivos para o seu constante crescimento de uso e desenvolvimento, a tendência da utilização do Sistema Operacional Linux.

Com tais fatos, os bibliotecários, que trabalham com a informação, como sua matéria-prima, é necessário que se adaptem com o processo de automação para melhoria dos serviços prestados pelas bibliotecas.

A automação do ambiente bibliotecário tem como principal objetivo oferecer ao usuário uma base de dados com informações de materiais adquiridos pelas escolas, de forma a facilitar seu acesso (Resende, 2000).

Se as bibliotecas escolares querem oferecer melhores serviços aos usuários e cumprir sua missão educacional, será necessário adaptar as tecnologias de acordo com às necessidades e quantidades de informação dispostas por meio da utilização de um SAB que atenda os recursos disponíveis sem se tornar obsoleto a longo prazo (Dias, 2010).

Diante disso, a biblioteca passa a assumir que deixará de ser considerada tradicional e passará a ser chamada de eletrônica. A biblioteca eletrônica é o termo utilizado que se refere ao sistema nos quais processos básicos da biblioteca são de natureza eletrônica (Marchori, 1997).

Isto implica a ampla utilização de computadores e de suas facilidades na construção de índices on-line, busca de textos completos e na recuperação e armazenamento de registros (Marchori, 1997).

Definir um SAB não é uma tarefa simples porque nos últimos 10 anos houve um grande avanço na área de desenvolvimento de sistemas, mas especialmente em softwares para automação de bibliotecas (Corte et. al, 1999).

A escolha de um software exige, fundamentalmente, a análise das ferramentas disponíveis, funções, recursos, potencialidade e a parceria com a empresa fornecedora, ou seja, existem várias funções para cada tipo de biblioteca, já que as formas de trabalho e prestação de serviços variam de uma para outras (Côrte, et. al, 1999).

Malheiros (2013), analisa que o processo de informatização das bibliotecas no Brasil passou a viver uma nova fase, caracterizada pela disponibilidade de recursos avançados – máquinas e softwares de nova geração que chegaram ao país nos últimos anos - ,apontando como causa, tanto de defasagem como do novo surto de desenvolvimento.

Malheiros (2013) alerta para o fato de que não existe um sistema ideal e, mesmo que a escolha seja a mais acertada, poderá não atender completamente os requisitos funcionais e de *perfomance*, além de executar o backup desejado e as opções de proteção, a custo compatível com o orçamento disponível.

Por esse motivo é importante que a biblioteca determine seus próprios requisitos obrigatórios e solicite as operações desejáveis somente após certificar-se de que as funções básicas e necessárias estejam plenamente atendidas.

Qualquer iniciativa de informatização de uma biblioteca ou centro de documentação deve, primeiramente, identificar a cultura, missão, objetivos e

programas de trabalho da organização; as características essenciais da biblioteca com relação a sua abrangência temática, serviços e produtos oferecidos; interesses e necessidades de informação dos usuários.

2.1 Openbiblio

O Openbiblio é o software livre para administração de bibliotecas mais utilizado no Brasil. O Sistema é capaz de gerenciar com facilidade os processos de uma biblioteca que incluem administração, relatórios, cadastro e circulação.

Uma das suas principais características é seguir os padrões internacionais US MARC - padrão internacional para transferência de dados entre sistemas de bibliotecas (SLB, 2010).

Por ser criado em PHP(*Hypertext PreProcessor*) e ter sua base de dados em MySQL(*Structured Language Query*), o sistema permite ser acessado através de um navegador web. Com isso pode ser implantado de forma centralizada diretamente de um servidor web, podendo ter acesso de qualquer lugar do mundo, ou então instalado localmente em um computador rodando Linux ou Windows (SLB, 2010).

A Figura 1 apresenta a tela do OpenBiblio, com as opções de empréstimo, catalogação, administração, relatórios. Destaca-se o módulo de catalogação que permite adicionar e classificar o acervo.

The screenshot shows the OpenBiblio cataloging module. At the top, there's a navigation bar with tabs: Inicio, Préstamo, Catalogación, Administración, Informes, and a Salir button. On the left, there's a sidebar with links: Búscar material, Nuevo material, Subir datos Marc, and Ayuda. The main area is titled 'Añadir nuevo Bibliografía:' and contains a form with the following fields:

Los campos marcados con * son requeridos.	
* Tipo de material:	audio tapes
* Colección:	audiolibro infantil
* Número de entrada:	<input type="text"/>
Mostrar en OPAC:	<input checked="" type="checkbox"/>
Campos USMarc:	
* Title:	<input type="text"/>
Remainder of title:	<input type="text"/>
Statement of responsibility, etc.:	<input type="text"/>
* Personal name:	<input type="text"/>

Figura 1 - Módulo de catalogação do OpenBiblio.

2.2 Biblivre

Em 2005, a empresa Sociedade dos Amigos da Biblioteca Nacional sob a presidência de Paulo Marcondes Ferraz, propôs o projeto de desenvolvimento da versão de um conjunto de programas de computador conhecido como Biblivre, seu propósito era tentar informatizar bibliotecas dos mais variados portes e tão somente propiciar a comunicação entre elas.

Nesse sentido, desde o início o desenvolvimento do sistema foi previsto que este software fosse oferecido livremente às bibliotecas que desejam utilizá-lo, estudá-lo, modificá-lo de acordo com os termos de licença que o caracterizassem como software livre. Devido a esta característica, o projeto passou a se chamar Biblioteca Livre (Alauzo et. al, 2014).

Este sistema possui os módulos de busca (permite consultar leitores e livros), circulação (permite saber se o exemplar está disponível ou não no acervo), catalogação (permite consultar, cadastrar, alterar e excluir um livro), aquisição (permite registrar outros materiais adquiridos) e administração (permite cadastrar usuários do sistema).

A Figura 2 apresenta a tela do Biblivre, com as opções de busca, circulação, catalogação, aquisição, administração e ajuda, podendo ser utilizado em 3 idiomas.



Figura 2 - Tela de cadastro de leitores no BibLivre.

2.3 Pergamum

O sistema integrado de bibliotecas Pergamum é um software informatizado de gerenciamento de bibliotecas que foi desenvolvido por profissionais da área de processamento de dados em parceria com bibliotecários, da Pontifícia Universidade Católica do Paraná (PUC-PR). Ele contempla as principais funções de uma biblioteca, funcionando de forma integrada, com o objetivo de facilitar a gestão dos centros de informação, bem como melhorar a rotina diária do seus usuários (Pergamum, 2013).

De acordo com Anzolin (2009, p. 47) sua estrutura está dividida em nove módulos: Módulos de parâmetros, estatísticas, aquisição, catalogação, circulação, relatórios, estatísticas, consulta ao catálogo e internet.

O uso do Pergamum trouxe um grande avanço na tecnologia da informação para o trabalho em rede, pois facilitou o acesso à informação, visto que permitiu ao usuário e operador consultar o acervo das bibliotecas da rede e, em uma única busca.

Entre as suas principais vantagens destaca-se o módulo Web, por disponibilizar o catálogo em formato eletrônico, apresentando recursos facilitadores no qual o usuário obtém prontamente as informações sobre o livro e/ou autor desejado. Além disso, a pesquisa é de fácil acesso sendo possível recuperar a informação por qualquer identificador da obra: autor, título, número de registro entre outros.

A Figura 3 mostra o módulo de pendências o qual verifica os livros que os usuários possuem e não foram devolvidos. Além disso mostra quantas renovações é permitido o usuário efetuar.

The screenshot shows the Pergamum library management system interface. At the top, there's a header with the logo of UFTPR (Universidade Tecnológica Federal do Paraná), the title 'Meu Pergamum', and user information like 'Seja bem-vindo(a)!'. On the right, there are links for 'Logout', 'Alto contraste', and 'Ajuda'. A sidebar on the left titled 'Servicos' includes links for 'Emprestimo', 'Perfil de interesse', 'Solicitações', 'Comentarios', 'Dados pessoais', and 'Alteração de senha'. The main content area has tabs for 'Informativo', 'Títulos pendentes', 'Títulos reservados', and 'Cesta Permanente'. The 'Títulos pendentes' tab is active, displaying a table of pending titles:

	Data Devolução	Nº Renovações / Limite	Multa parcial (R\$)
Desenvolvendo aplicações web com JSP, Servlets, JavaServer faces, Hibernate, EJB 3 persistência e AJAX / 2007 - Livros	09/06/2017	0 / 3	Renovar 0
Java Web Services : implementando / 2010 - Livros	09/06/2017	0 / 3	Renovar 0
PostgreSQL : guia do programador / 2008 - Livros	05/06/2017	0 / 3	Renovar 0
SQL : um guia para iniciantes - 3. ed. / 2009 - Livros	05/06/2017	0 / 3	Renovar 0

At the bottom of the page, there's a footer with the text '© 2000 - 2014. Pergamum. Todos os direitos reservados.'

Figura 3 Tela de pendências do meu pergamum por usuários.

2.3.1 Implantação do pergamum na rede de bibliotecas da prefeitura de Vitória-ES

Segundo a Prefeitura de Vitória (2012) a rede pergamum é considerada uma das melhores do Brasil, sendo referência para todo país no que se refere à prestação de serviços no âmbito bibliotecário.

As bibliotecas que fazem parte dessa rede ainda não possuem um sistema gerenciador de serviços biblioteconômicos, fato que estimulou os profissionais bibliotecários a solicitarem junto a secretaria de educação, a automação das bibliotecas por meio de um sistema seguro e efetivo.

O processo de implantação do sistema de automação e administração de bibliotecas escolares se inicia com o processo de articulação na Secretaria de Educação em função do diagnóstico realizado juntos às bibliotecas das Escolas Municipais de Ensino Fundamental. Tal análise vem sendo realizado desde o processo inicial de revitalização dos espaços escolares no ano de 1999. Desde então, os bibliotecários atuantes na rede vêm apontando a necessidade eminente de aquisição de um sistema de automação e administração de bibliotecas, pois várias tentativas de aquisição e desenvolvimento de um sistema já foram feitas.

3. Fundamentação Teórica

Nesse capítulo será abordado um pouco sobre as tecnologias que serão utilizadas para desenvolver esse projeto.

3.1 Arquitetura Cliente-Servidor

Em uma arquitetura cliente-servidor há um hospedeiro sempre em funcionamento, denominado servidor, que atende a requisições de muitos outros hospedeiros, denominado clientes.

Um exemplo clássico é a aplicação web na qual um servidor web que está sempre em funcionamento atende a requisições de *browsers* de clientes. Quando

recebe uma requisição de um objeto de um cliente, um servidor web responde enviando o objeto requisitado a ele. Observa-se que nesta arquitetura, os clientes não se comunicam diretamente uns com os outros. Outra característica dessa arquitetura é que o servidor tem endereço IP (*Internet Protocol*) fixo.

Em aplicações cliente-servidor, muitas vezes acontece de um único servidor ser incapaz de atender todas as requisições dos clientes, por esse motivo um grande conjunto de hospedeiros chamado *data center* frequentemente é usado para criar um servidor virtual em arquiteturas cliente-servidor (Kurose, 2010).

A Figura 4 mostra de forma ilustrativa como está organizada a arquitetura cliente/servidor.

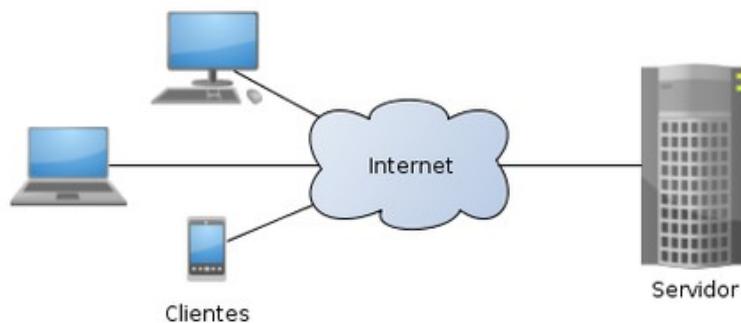


Figura 4 Como funciona a arquitetura cliente-servidor

3.2 Java Enterprise Edition – JEE

É uma plataforma *open-source* amplamente usada que contém um conjunto de tecnologias que reduz a complexidade do desenvolvimento, implantação e gerenciamento de aplicações de várias camadas centradas no servidor (ORACLE, 2013).

Essa tecnologia consiste em um sistema de multcamadas que dá suporte a componentes de apresentação (Java Server Faces, Servlets), auxilia na utilização de regra de negócio e também na persistência no banco de dados como Enterprise Java Beans (EJB) (Paz, et. al, 2016).

Além disso, suporta conexões com o banco de dados, e possibilita flexibilidade da arquitetura do projeto de forma simples (Paz, et. al, 2016).

O JEE é construído sobre a plataforma Java SE (*Standart Edition*) e oferece um conjunto de APIs (*Application Programming Interface*) para desenvolvimento e execução de aplicações portáteis, robustas, escaláveis, confiáveis e seguras no lado do servidor (Oracle, 2013).

As APIs disponibilizadas pelo JEE para desenvolvimento de aplicações cliente-servidor são: Java Server Pages(JSP), Java Servlets, Java Server Faces (JSF), Enterprise Java Beans (EJB), Java Persistence (API). (objetos distribuídos, cluster, acesso remoto a objetos).

3.2.1 Java Server Pages - JSP

Consiste em uma tecnologia que simplifica o processo de desenvolvimento de aplicações para web. A tecnologia interage fortemente com Java Hypertext Markup Language (JHTML), (Hackenhaar et. al, 2010).

O JSP incorpora elementos dinâmicos em páginas web usando Java embutido de algumas *tags* de marcação simples (Fields, 2000).

3.2.2 Java Servlets

Classes Java desenvolvidas que implementam um Servlet Container (um servidor que permita a execução de Servlets, chamado de servidor de aplicações Java), que podem tratar as requisições recebidas de clientes (Gonçalves, 2007).

3.2.3 Java Server Faces - JSF

Tecnologia para simplificar o desenvolvimento de aplicações Web através de componentes de interface de usuário (GUI) e conecta esses componentes a objetos de

negócios. Também automatiza o uso de JavaBeans e navegação de páginas (Gonçalves, 2007).

3.2.4 Enterprise Java Beans - EJB

Componente servidor que roda em um contêiner do servidor de aplicação. Fornecer um rápido e simplificado desenvolvimento de aplicações Java baseada em componentes, distribuídas, transacionais, seguras e portáveis (Gonçalves, 2007).

3.2.5 Java Persistence API - JPA

A JPA é baseada no conceito de Plain Old Java Object, que incorpora ideias de renomados *frameworks* de persistência para padronizar o mapeamento O/R em Java.

3.3 Apache Tomcat Server

É um servidor de aplicação Java para Web, mais especificamente, um contêiner de servlet. Desenvolvido pela Apache Software Foundation, é distribuído como software livre sendo oficialmente endossado pelo Sun como a implementação de referência para as tecnologias JavaServlet e JSP.

Robusto e eficiente para ser utilizado num ambiente de produção. Tem a capacidade de ser de atuar também com servidor web, ou pode funcionar integrado a um servidor web. Como servidor web ele provê um servidor Web HTTP puramente em Java (Carlos, 2008).

3.4 Servidor de banco de dados MySQL

É um servidor de banco de dados relacional que se aproxima cada vez mais das facilidades em produtos proprietários, independente do sistema operacional que for utilizado (Gilmore, 2011).

O MySQL oferece diversos tipos de mecanismos para gerenciamento de dados, conhecidos como mecanismo de armazenamento. Cada mecanismo ostenta seus próprios pontos fortes e fracos, e devem ser aplicados de forma seletiva para acomodar o melhor uso que se espera para os seus dados (Gilmore, 2011).

Além disso permite a indexação e busca *full-text* como um recurso que melhora o desempenho da mineração de dados a partir das colunas baseadas em texto. Com esse recurso é possível produzir resultados na ordem de relevância, de acordo com a proximidade com que a *query* casa as colunas de texto indexadas na linha (Gilmore, 2011).

3.5 Unified Modeling Language (UML)

A UML é uma linguagem para especificação, documentação, visualização e desenvolvimento de sistemas orientado a objetos. Sintetiza os principais métodos por meio de diagramas que representam sistemas de softwares sob diversas perspectivas de visualização. Facilita a comunicação entre todas as pessoas envolvidas no processo de desenvolvimento de um sistema por apresentar um vocabulário de fácil entendimento (OMG, 2005).

Alguns dos diagramas utilizados da UML são

1. Diagrama de classes.
2. Diagrama de caso de usos.
3. Diagrama de sequência.
4. Diagrama de estados

3.5.1 Diagramas de casos de uso

O diagrama de casos de uso tem como objetivo especificar todas as funcionalidades do sistema que será desenvolvido, estas funcionalidades são os requisitos levantados a partir de entrevistas. Esse diagrama é composto por três elementos (atores, casos de usos e relacionamentos).

Alguns relacionamentos do diagrama de casos de uso:

1. Associação : Demonstra quais atores utilizam quais funcionalidades do sistema.

2. Generalização : Ocorre em casos de usos distintos possuem características semelhantes
3. Extensão : Quando um caso de uso é ligado a outro, porém sua execução é opcional.
4. Inclusão : Um caso de uso depende da execução do outro.

O objetivo é estruturar o sistema com todas as funcionalidades que deverá conter e quais atores farão o uso de certas funcionalidades (Dias et al., 2015)

3.5.2 Diagrama de classes

É um modelo fundamental de uma especificação orientada a objetos. Produz a descrição mais próxima a estrutura do código de um programa, ou seja, mostra o conjunto de classes com seus atributos, métodos e os relacionamentos entre elas (Vargas, 2008).

3.6 Framework - Hibernate

É um projeto que oferece uma solução para o problema de gerenciamento de dados persistentes em Java. Se relaciona com o banco de dados, onde esse relacionamento é conhecido como mapeamento objeto/relacional para java, deixando o desenvolvedor livre para se concentrar em problemas da lógica do negócio.

Sua simplicidade em configuração, dá ao desenvolvedor algumas regras para que sejam seguidas como padrões de desenvolvimento ao escrever sua lógica de negócio e suas classes persistentes (Gonçalves, 2007).

O Hibernate disponibiliza para seus usuários um mecanismo de consulta de dados, que posteriormente diminui o tempo de desenvolvimento que seria gasto em uma determinada aplicação. Por utilizar sua própria linguagem de consulta o Hibernate Query Language (HQL) disponibiliza a função de consulta e retorno de dados, fazendo a conversão entre os registros permitindo que a aplicação utilize qualquer banco de dados SQL (Magalhães et. al, 2013).

4. Procedimentos Metodológicos/Metodologia

A abordagem metodológica que será utilizada por este trabalho é a *Rational Unified Process* (RUP) e compreende as seguintes etapas:

1. Concepção – esta etapa identifica e descreve quais recursos e funções compõem o sistema. Além disso propõe uma arquitetura com esquema provisório dos principais sub-sistemas e da função dos recurso que a compõem (Pressman, 2011).

2. Elaboração – envolve as atividades de obtenção e análise de requisitos. Refina e expande as informações da etapa de concepção realizando o levantamento de requisitos funcionais e não funcionais assim como a construção dos modelos de análise. Além disso inclui a modelagem rápida do projeto que envolve a estruturação do banco de dados e dos módulos do sistema. Esta etapa demonstra a viabilidade da arquitetura mas não oferece os recursos e funções necessárias para usar o sistema (Pressman, 2011).
3. Construção - Compreende a codificação e o desenvolvimento de protótipos dos módulos do sistema com integração com o banco de dados. Modelos de requisitos e de projeto iniciados durante a etapa de elaboração, são completados nesta etapa para refletir as versões do software. Implementa recursos e funções necessárias do sistema em uma linguagem de programação. À medida que os componentes estão sendo implementados, desenvolve-se e executam-se testes de unidades para cada um deles. Além disso, realizam-se atividade de integração (montagens de componentes e teste de integração) de todas as funcionalidades dos módulos do sistema e do banco de dados (Pressman, 2011).
4. Validação – Compreende os últimos estágios da atividade da construção do sistema e a primeira parte da atividade de emprego genérico. Valida-se o software com usuários finais para testes beta e o *feedback* dos usuários relataram defeitos e mudanças necessárias (Pressman, 2011).

5. Desenvolvimento Preliminar

Como etapa da abordagem metodológica da RUP, iniciou-se a concepção do sistema. Para especificar os requisitos foi realizada uma visita técnica na biblioteca do Colégio Estadual Newton Felipe Albach com o objetivo de entrevistar o funcionário responsável e questionar como são gerenciadas as atividades do dia-dia. A seguir é mostrada a especificação de requisitos obtida.

5.1 Especificação de requisitos

Atualmente a biblioteca não possui um sistema informatizado que gerencie o acervo. Isto é devido por que os programas utilizados anteriormente eram aplicações desktop vulneráveis ao ataque de vírus e dependentes dos recursos locais do hardware do computador.

Cenário 1 - Concessão e controle de empréstimos

Cada turma tem um dia da semana e um horário específico para ir até a biblioteca para solicitar empréstimo ou renovar livros. Os empréstimos são controlados de forma manual por meio de um formulário individual de turmas que é impresso semanalmente que contem a data, nome do aluno, título do livro, assinatura do aluno, devolução.

O procedimento de renovação é realizado por meio de uma marca no formulário em forma de bola ao lado do nome do aluno. O procedimento de devolução é realizado também nesse mesmo formulário colocando um “ok”.

Em caso de perda ou danificação de um livro o aluno fica bloqueado de empréstimo até o momento que doar um livro igual ao perdido ou danificado.

Problemas no cenário

- Dificuldades de controle do acervo.
- Falta de recursos financeiros para adquirir e manter um sistema informatizado.
- Falta de exemplares de livros muito desejado pelos alunos.
- Livros que não existem mais fisicamente, mas existem no sistema.

Cenário 2 - Funcionamento da biblioteca

O ambiente da biblioteca tem apenas um funcionário. A catalogação é realizada à partir de um sistema que fornece um número aleatório que não é gerado novamente.

A biblioteca funciona de manhã das 7:30 às 12:00, a tarde das 13:00 às 17:30 de segunda-feira à sexta-feira, sendo fechada no período noturno, férias escolares, fins de

semana e feriados. Hoje em dia a escola possui 407 alunos, todos com acesso aos livros.

Problemas no cenário

- Funcionário único com sobrecarga de trabalho (o funcionário não trabalha somente na biblioteca).
- Apenas um computador antigo na biblioteca e com recursos limitados (onde acessam alunos, professores e o bibliotecário).

Cenário 3 - Classificação do acervo

Os livros são classificados em oito gêneros literários distintos identificados por uma etiqueta colorida na lateral. Essa etiqueta auxilia o aluno na hora da escolha do livro, pois os livros não estão organizados por gênero na prateleira e sim por ordem alfabética.

As cores vinculadas com o gênero de livros são: Laranja – Contos, Azul – Poesia, Vermelho – Crônicas, Amarelo – Literatura Infanto-Juvenil, Roxo – Literatura Juvenil, Preto – Suspense, Vinho – Narrativas, Azul-Escuro – Teatro.

Cenário 4 - Controle dos livros didáticos

Livros didáticos e DVDs também estão na biblioteca mas não possuem classificação. Os livros didáticos são renovados há cada quatro anos. São divididos por matéria, série, e Programa Nacional do Livro Didático (PNLD).

O controle dos livros didáticos é realizado de forma manual e registrado em um formulário que possui o ano letivo, a turma, números de alunos, nome do aluno, assinatura do aluno, disciplina(Arte, Biologia, Filosofia, Física, Geografia, História, Matemática, Português, Química, Sociologia, Inglês).

Quando há livro didático disponível para todos os alunos, o aluno fica com o livro em posse durante o todo ano letivo, caso pedir transferência ele terá que devolver os livros pendentes a escola.

Quando não há livro didático para todos os alunos, o livro é usado durante o período de aula, assim o aluno deverá devolver ao término da aula.

Cada disciplina possui dois campos: Recebido (R), Devolvido (D), os quais são marcado com “x” quando a ação é realizada.

Problemas no cenário

- Dificuldades de controle dos livros didáticos.
- Transferências (Certos alunos vão para nova escola sem devolver os livros).
- Irresponsabilidade por parte do aluno (ele não perde o livro, porém ele não entrega no fim do ano letivo).

Cenário 5 - Atrasos

Para alunos que possuem atraso na entrega de livros, não é cobrado multa em dinheiro, mas existe uma penalidade na qual consiste em impedir o empréstimo de outros livros enquanto estiver com livros atrasados. Caso acontecer do aluno emprestar um livro sete dias antes de um feriado e não renovar, ele poderá devolver no dia letivo após o feriado sem ser considerado atraso. Se o aluno faltou no dia devolução ele poderá entregar no dia seguinte justificando o motivo da falta sem ser considerado atraso.

Problemas no cenário

- Faltas sem justificativa.
- Feriados durante a semana (ás vezes longos).
- Suspensão de aulas em dias letivos (greves, atividades pedagógicas, fenômenos da natureza).

Cenário 6 – Dívidas

Por meio do formulário de empréstimos da semana anterior, são verificados os alunos que não devolveram o livro ou não renovaram. A partir desses dados é criado um formulário com a relação de alunos com pendências de livros. Este formulário possui o ano, turma, nome do aluno, livro e a data do empréstimo.

Problemas no cenário

- Alunos que passam a férias com o livro e não devolve após as férias.
- Falta de controle de alunos.
- Transferências (certos alunos querem tirar vantagem que pelo fato de ser transferido, não precisam devolver os livros pendentes a escola a qual pediu a transferência).

5.2 Requisitos funcionais do sistema

1.Módulo de livros literários – A partir desse módulo será cadastrado todo acervo pertencente a biblioteca. Sendo composto pelo campo nome, autor, gênero, número de páginas, número que o livro foi catalogado.

2.Módulo de livros didáticos – A partir desse módulo serão cadastrados todos os livros didáticos da escola. Sendo composto pelo campo código, título, disciplina, aluno, série responsável pelo livro, ano, PNLD (Programa Nacional do Livro Didático).

3.Módulo de turmas - Nesse módulo será cadastrado todas as turmas de uma escola. Sendo composto pelos campos: turmas e dia da semana, horário de emprestar livro, professor de português.

4.Módulo de alunos - Neste módulo será cadastrado todos os alunos da escola. Composto pelos campos: nome, e-mail, celular, turma, nome do pai, nome da mãe.

5 - Módulo de empréstimos - esse módulo será composto pelos campos : aluno, data que foi efetuado o empréstimo e também haverá um sub-módulo para incluir os livros

no empréstimo, data prevista para devolução, somente pessoas que fazem parte da comunidade escolar podem efetuar o empréstimo, se um pai ou mãe de aluno emprestar um livro o empréstimo ficará no nome do filho.

6 - Módulo de devoluções - esse módulo será composto pelos campos livro, empréstimo, data que foi efetuada a devolução, justificativa por que não entregou o livro, e um sub-módulo para incluir os livros devolvidos, data que deveria ser devolvido, caso a data de devolução for posterior a data prevista da devolução o sistema deverá penalizar o aluno de realizar empréstimos enquanto estiver com livros em atraso.

7 - Calculo da devolução - será somado 7 dias após a data do empréstimo. (desconsiderando feriados e recessos).

8 - Segurança por meio de sessões de login - Os usuários poderão ter acesso as funcionalidades do sistema somente se estiver logado por meio de uma sessão, como não tem restrições de acesso no sistema todo os funcionários que trabalham na biblioteca poderão usar com um usuário e uma senha padrão.

9 -Suspensão do empréstimo – Para os alunos que estiverem perdido ou danificados livros da biblioteca ou também que possuem livros em atraso, o empréstimo deverá ser bloqueado enquanto o aluno não repor o livro ou devolver todos os que estiverem com atraso.

10. Relatório de alunos com livros atrasados – Deve ser exibido um relatório de alunos com livros atrasados contendo o nome, turma, livros, celular, e-mail.

5.3 Requisitos não-funcionais do sistema

1. Relatórios mensais de empréstimos – Serão emitidos relatórios de empréstimos agrupados por meses.

5.4 Principais características e funções do sistema

- Arquitetura cliente-servidor.
- Estruturado por vários módulos.
- Módulos vinculados com as propriedade *Create, Read, Update, Delete (CRUD)*.
- Módulos para automatizar os principais processos de uma biblioteca.

6. Resultados esperados

A partir dos cenários da especificação de requisitos, apresentam-se os diagramas UML de casos de uso que determinam a funcionalidade e as características do software a ser desenvolvido.

Caso de uso - Cenário 1 : concessão e controle de empréstimo

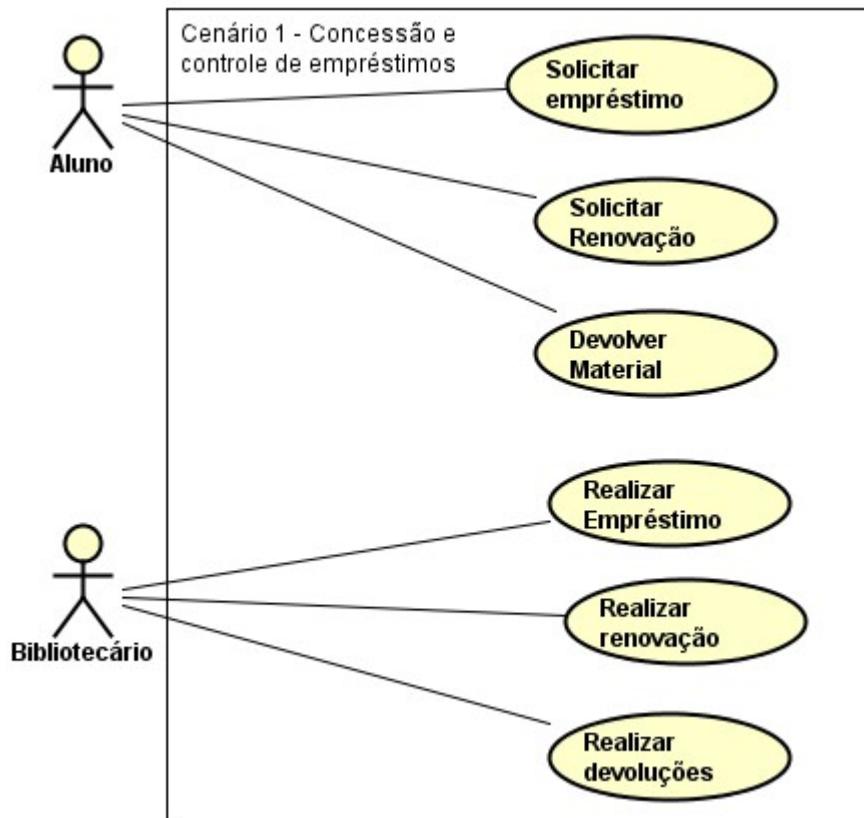


Figura 5. Diagrama de casos de uso que modelam o cenário 1 (concessão e controle de empréstimos)

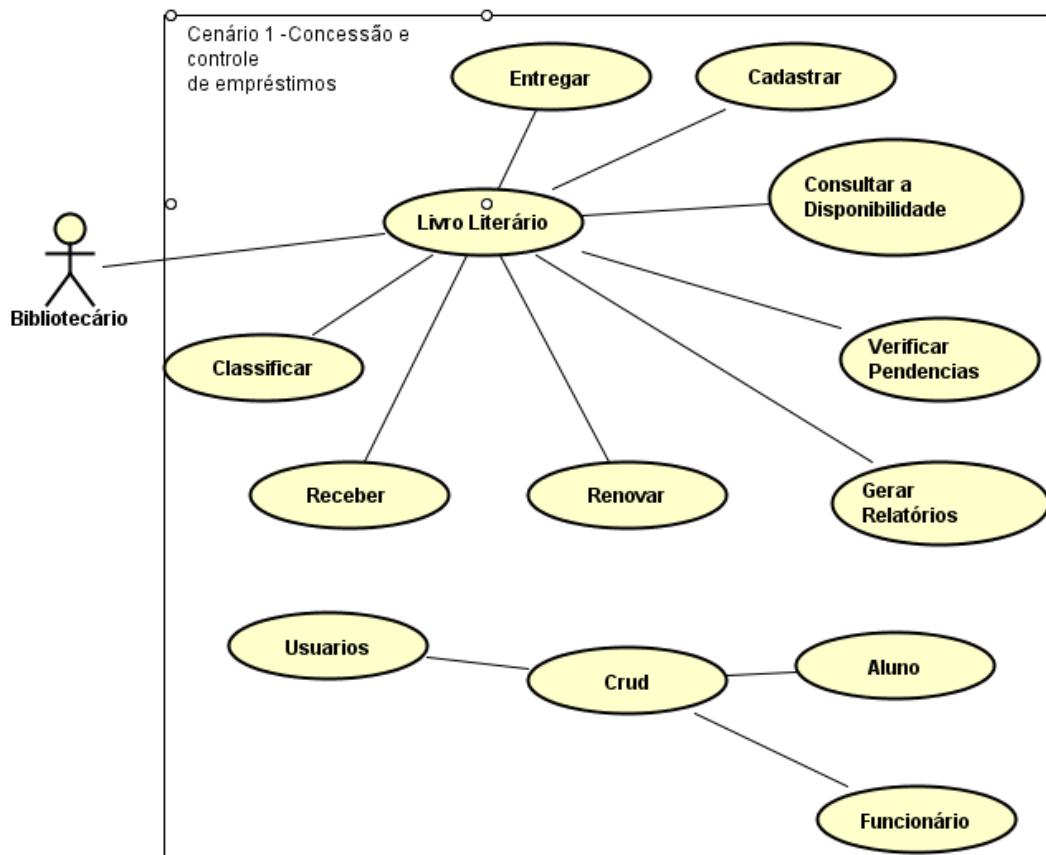


Figura 6. Diagrama de casos de uso que modelam o cenário 1 desde o ponto de vista do ator bibliotecário.

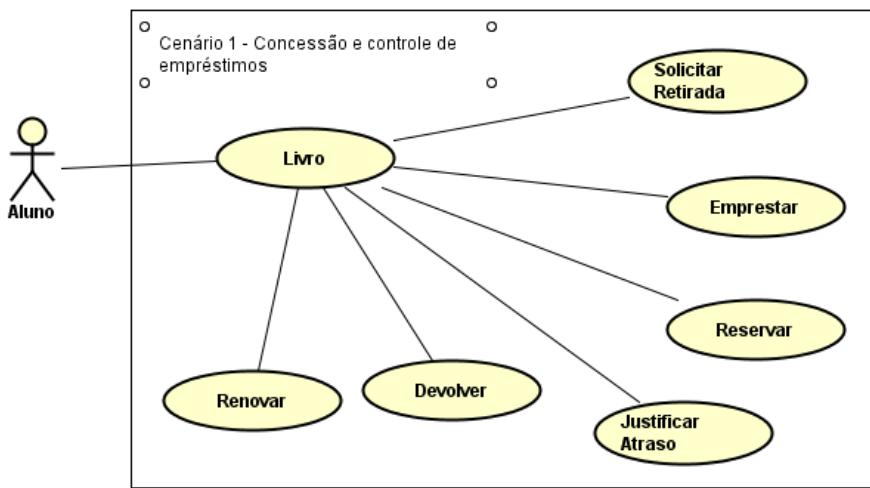


Figura 7. Diagrama de casos de uso que modelam o cenário 1 desde o ponto de vista do ator aluno

Caso de uso - Cenário 2 : Funcionamento da biblioteca

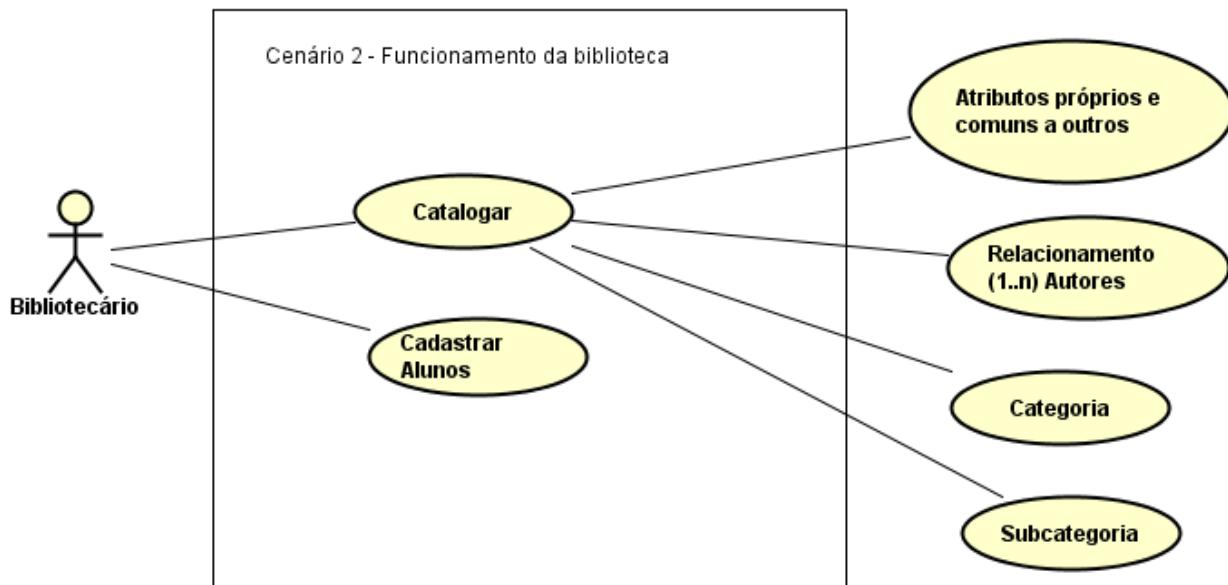


Figura 8. Diagrama de casos de uso que modelam o cenário 2 (Funcionamento da biblioteca)

Caso de uso - Cenário 3 : classificação do acervo

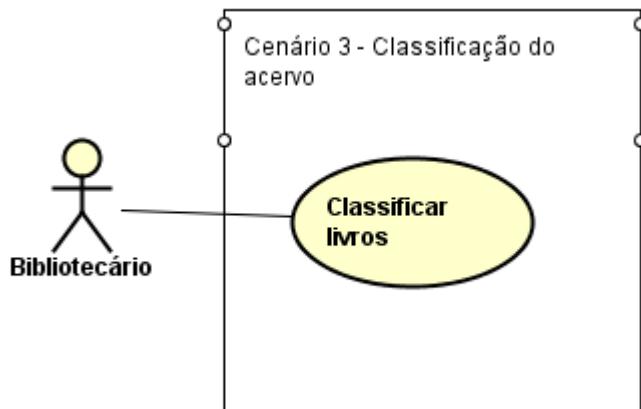


Figura 9. Diagrama de casos de uso que modelam o cenário 3 (classificação do acervo).

Caso de uso - Cenário 4 : controle de livros didático

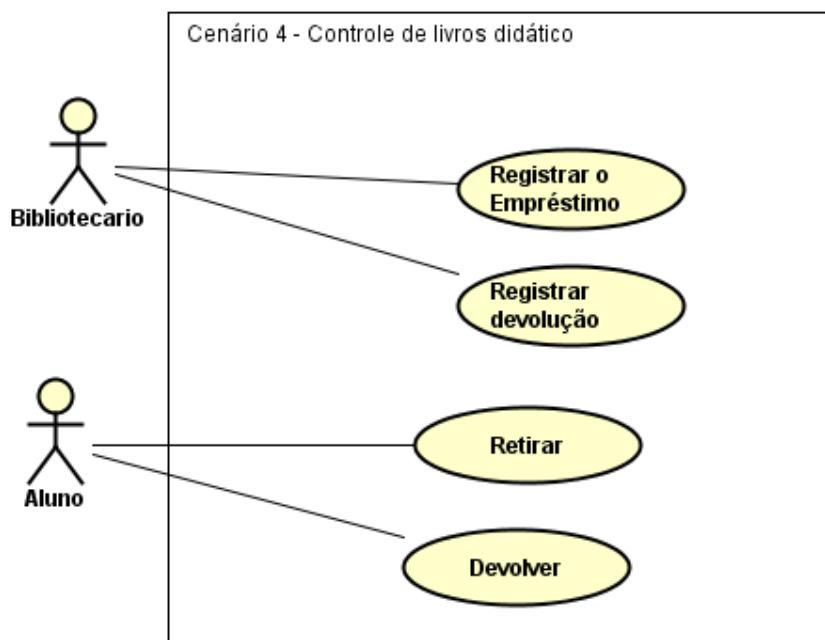


Figura 10. Diagrama de casos de uso que modelam o cenário 4 (controle de livros didáticos).

Caso de uso - Cenário 5 : atrasos

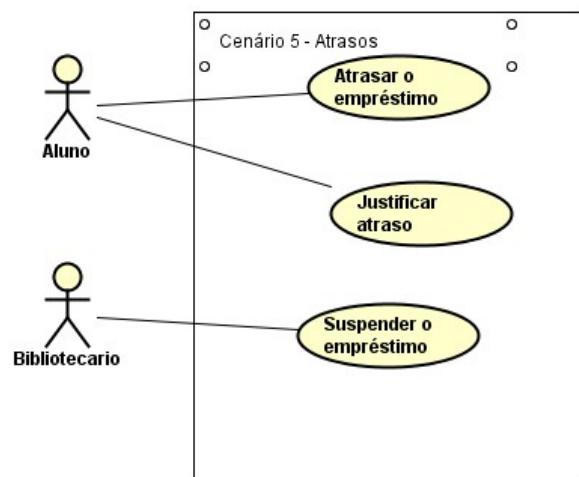


Figura 11. Diagrama de casos de uso que modelam o cenário 5 (atrasos).

Caso de uso - Cenário 6 : dívidas

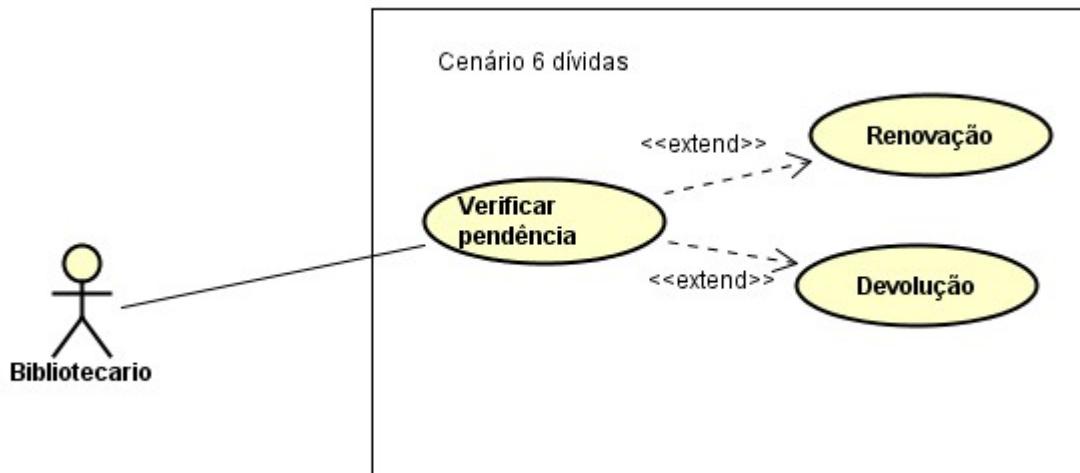


Figura 12. Diagrama de casos de uso que modelam o cenário 6 (dívidas).

7. Considerações Finais

O objetivo desse trabalho é desenvolver um sistema web que permita o acesso em diferentes máquinas de uma mesma escola sem a necessidade de instalar o sistema em cada máquina. Em um primeiro momento esse trabalho será testado na biblioteca Colégio Estadual Newton Felipe Albach, caso outras escolas da rede pública de Guarapuava tiver interesse neste trabalho ele poderá ser implantado.

Por meio da visita técnica realizada na biblioteca e da entrevista com o funcionário, percebe-se que o controle de livros didáticos, empréstimos e atrasos de livros ainda é feito em formulários de papel, que são impressos.

Dessa mesma entrevista conseguiu-se modelar e definir quais serão os módulos do sistema. O desenvolvimento desse sistema irá contribuir para facilitar o controle de saídas e entradas da biblioteca.

Muitos colégios da rede pública de Guarapuava ainda não informatizaram suas bibliotecas, pela falta de recursos financeiros. A partir desse fato foi inspirado o desenvolvimento desse projeto. Colégios localizados em bairros pobres, distantes do centro poderão ser ou serão os últimos a contar com esse recurso.

Destacam-se algumas dificuldades encontradas para o desenvolvimento como, a falta de contato com alguns sistemas bibliotecários, material escasso de estudo das tecnologias a serem utilizadas e tempo curto para o desenvolvimento do projeto

8. Cronograma

Atividades	TCC 1						TCC 2				
	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez	
1. Revisão dos apontamentos da banca		X		X			X				
2. Revisão bibliográfica	X	X	X	X	X	X					
3. Levantamento das tecnologias para o desenvolvimento			X	X	X						
4. Concepção				X							
4.1 levantamento de requisitos				X							
4.2 identificar requisitos funcionais ou não funcionais		X	X								
5 Elaboração				X	X	X					
5.1 escolher as tecnologias a serem utilizadas				X							
5.2 modelagem do banco de dados					X	X					
5.3 modelagem dos módulos que componhem o sistema	X	X									
6 construção						X	X	X			
6.1 telas do sistema						X					
7 Validação									X		
7.1 Testes									X		
7. Redação do projeto de TCC.	X	X	X	X							
8. Defesa do projeto de TCC				X							
9. Escrita da Monografia de TCC					X	X	X	X	X		
10. Elaboração da apresentação final.								X	X		
11. Defesa final do TCC									X		

Quadro 1 – Cronograma de desenvolvimento do projeto.

9. Referências

PRESSMAN, Roger S. Engenharia de Software Uma Abordagem Profissional. 7º edição. Porto Alegre : AMGH , 2011.

GILMORE, W. Jason. Dominando PHP e MySQL. 2ª edição. Rio de Janeiro : Alta Books, 2011.

GONÇALVES, Edson. Desenvolvendo aplicações web com jsp, servlets, jsf, hibernate, ejb 3 persistence e ajax. Rio de Janeiro : Ciência Moderna, 2007.

KUROSE, James F. Redes de computadores e a Internet. 3ª edição. São Paulo : Person, 2010.

MALHEIROS, Tania Milca de Carvalho. **NECESSIDADE DE INFORMAÇÃO DO USUÁRIO COM DEFICIÊNCIA VISUAL: UM ESTUDO DE CASO DA BIBLIOTECA DIGITAL E SONORA DA UNIVERSIDADE DE BRASÍLIA.** 305 f. Dissertação de Mestrado – Universidade de Brasília, Brasília, 2013.

ALAUZO, Jorge Luiz Cativo; SILVA, Danielle Lima; Fernandes, Tatiana Brandão. FUNCIONALIDADES DE UM SOFTWARE LIVRE DE AUTOMAÇÃO DE BIBLIOTECAS: uma avaliação do Biblivre. **RacIn**, João Pessoa, v.2, n.2, p. 23-43, jul.-dez 2014.

CORTE, Adelaide Ramos; ALMEIDA, Ieda de Muniz; PELLEGRINI, Ana Emilia; LOPES, Ildeu Ordini; SAENGER, José Carlos; ESMERALDO, Maria Bernadete P.; PEREIRA,

Maria Cristina Moraes; FERREIRA, Rosana Rika M. C.; do LAGO. Automação de bibliotecas e centros de documentação: o processo de avaliação e seleção de softwares. Ci. Inf., Brasília, p.241-256, set-dez 1999.

PAZ, Marcelo Augusto Lobo; Dias, Jayme Willian. DESENVOLVIMENTO WEB UTILIZANDO JAVA EE, UNIPAR, Paranavaí, 2016.

MAGALHÃES. Carlos Filipe; Silva, Alexandre Paulino Sierra. Utilização do Framework Hibernate em Aplicação Java Web, UNIPAR, Paranavaí, 2013.

Viva o Linux. **Instalando Apache Tomcat em Linux 2.6.**

Disponível em <<https://www.vivaolinux.com.br/artigo/Instando-Apache-Tomcat-em-Linux-2.6>>. Acesso em: 05 de junho de 2017.

SLB – Software Livre do Brasil. **A nova versão do Openbiblio 0.6 pt.BR já está disponível.** Disponível em <<http://softwarelivre.org/portal/noticias/a-nova-versao-do-openbiblio-0.6-pt.br-ja-esta-disponivel1>>. Acesso em 30 de abril de 2017.

SILVA, Marcelo Bezerra; DIAS, Guilherme Ataíde. O sistema de automação em bibliotecas OpenBiblio aplicado à disciplina automação em bibliotecas. Biblionline, João Pessoa, v. 6, n. 1, p. 53-71, 2010.

Vargas, Thania Clair de Souza. A história da UML e seus diagrama. UFSC, Florianópolis, 2008.

